Models and algorithms for integrated production and distribution problems

Published Version:
Bo V., Bortolini M., Malaguti E., Monaci M., Mora C., Paronuzzi P. (2021). Models and algorithms for integrated production and distribution problems. COMPUTERS & INDUSTRIAL ENGINEERING, 154, 1-11 [10.1016/j.cie.2020.107003].

Availability:
This version is available at: https://hdl.handle.net/11585/802087 since: 2024-02-28

Published:
DOI: http://doi.org/10.1016/j.cie.2020.107003

Vincenzo Bo, Marco Bortolini, Enrico Malaguti, Michele Monaci, Cristina Mora, Paolo Paronuzzi, Models and algorithms for integrated production and distribution problems, Computers & Industrial Engineering, Volume 154, 2021, 107003, ISSN 0360-8352

The final published version is available online at:

https://doi.org/10.1016/j.cie.2020.107003

DOI 10.1016/j.cie.2020.107003

Models and Algorithms for Integrated Production and Distribution Problems

Vincenzo Bo[1], Marco Bortolini[2], Enrico Malaguti[3], Michele Monaci[3], Cristina Mora[2], Paolo Paronuzzi [3]

[1] *Plannet S.r.l., Via Pier Carlo Cadoppi 4/6, I-42124 Reggio nell'Emilia, Italy*
[2] *DIN, Università di Bologna, Viale Risorgimento 2, I-40136 Bologna, Italy*
[3] *DEI, Università di Bologna, Viale Risorgimento 2, I-40136 Bologna, Italy*
vincenzo.bo@plannet.it, {marco.bortolini3, enrico.malaguti, michele.monaci, cristina.mora, paolo.paronuzzi}@unibo.it

December 17, 2019

Abstract

Motivated by a real-world application, we consider an optimization problem that involves production, distribution and warehouse logistics. We present a mathematical formulation of the problem based on a Mixed Integer Linear Programming (MILP) model, and a metaheuristic algorithm which we use to compute approximate solutions. We test the proposed algorithms on two real-world test-cases and on a large set of realistic problems. The results show that, in all cases, the algorithm is very fast and produces solutions whose quality is very close to those that can be obtained by running a state-of-the-art commercial solver on the mathematical model for a very long time.

Key words: production planning, Mixed Integer Linear Programming, heuristic algorithms, lot-sizing, multiple plants, computational experiments.

1. Introduction

Production and distribution are two critical activities in supply chain management. They consist on the coordinate set of actions allowing the match between industries and their markets, vendors and buyers, suppliers and purchasers. In the last years, optimisation modelling of integrated production and distribution plans has raised significant interest among both researchers and practitioners, see e.g., Fahimnia et al. [1]. With the increase of competition and globalization, their role shifted from operative to tactical and from a separate local vision to an integrated management of these functions. Nowadays, competition is among collaborative interconnected and optimized supply chain networks and the integration of location decisions with other decisions, such as the structure of the supply chain network, the sustainability, the reverse logistics, delivery time and operating costs, is relevant to the design of a supply chain network (see Melo et al. [2], and Bortolini et al. [3]). Each of them links, organically, companies to their markets and strategic suppliers.

Following this trend, production and distribution planning rises as an integrated decisional science to provide strategies and quantitative methods to fulfill the market demand, answering the research and industrial question "What to produce and deliver, when and how?". Behind such question, the following multiple dimensions of production and distribution planning emerge:

- dynamic dimension, i.e., time drives the decisions of producing and distributing according to the market due dates and order delivery time;
- system dimension, i.e., the supply chain configuration (layers, entities, routes, etc.) constraints and guides the decisions;
- goal dimension, i.e., the goals to pursue as efficiency, cost, service level, green and social aspects, etc;
- stakeholder dimension, i.e., the decision-making process is tailored on the interests of the supply chain actors.

The convergence of the introduced multiple dimensions is hard, making production and distribution planning a wide, complex and dynamic problem to solve. The scientific literature agrees to adopt an integrated and stepwise approach following the impact of decisions and the horizon of the plan. Strategic, tactical and operative clusters of decisions are thus introduced (see Maravelias and Sung[4]). The strategic level focuses on long-term decisions over a multiple-year time horizon. At this level, the key elements of the supply chain system are fixed, e.g., overall structure, plant capacity, shipping modes, etc. Generally, the temporal dimension is neglected, and a single large time window is used. The inclusion of the temporal aspect of supply chains is the focus of the tactical level, where a shorter planning time horizon, e.g., one or two semesters, is typically considered. At this level, decisions on capacity and production allocation among plants, network flows, shipping and storage modes are taken from an integrated perspective minimizing costs and optimizing the supply chain service level. Finally, daily scheduling and routing activities are among the most important decisions of the operative level.

This paper, driven by an industrial application, aims at contributing to the tactical cluster of decisions. The goal is to provide an analytical model and a solution approach to support planners for an integrated production and transportation system within a multi-product, multi-plant and multi-period context. An aggregated cost function includes relevant drivers affecting the product total cost of ownership, while production and distribution constraints link the model to the industrial practice. We present a solution approach based on heuristic and metaheuristic optimization algorithms. To assess its effectivity and capability to guarantee the expected level of performance (in terms of computing time), we computationally test the approach to full-scale real-world instances from industry and to a large benchmark of realistic instances.

The paper is organized as follows. In Section 2 we review the relevant literature on production and distribution planning, both in terms of supply chain management and for what concerns optimization algorithms. Section 3 gives a formal definition of the problem at hand and introduces a mathematical formulation based on a Mixed Integer Linear Programming (MILP) model. In Section 4 we present a fast heuristic algorithm based on a greedy approach. In Section 5 we propose a metaheuristic approach based on the ruin-and-recreate paradigm. Finally, in Section 6 we give the outcome of computational experiments for the proposed algorithms on two real-world test-cases and on a large set of realistic instances that are derived from the real ones. Conclusions are drawn in Section 7.

2. Literature review

According to the multiple dimensions of production and distribution planning, the recent literature presents a major set of contributions on models, approaches and applications. In this section we first review a large body of literature in order to identify the relevant patterns in problem statement. Then, we move to solution approaches and we focus on a more specific class of problems, whose underlying structure generalizes the classical lot-sizing problem.

Concerning the problem statement, relevant patterns to tailor production and distribution planning problem deal with (i) the physical entities included in the problem, (ii) the drivers of cost computing the planning performances and (iii) the operative constraints making the problem relevant and linked to the industrial practice.

For what concerns the first point (*entity dimensions*), most papers focus on the following aspects:

- the mix to produce and distribute, i.e., single- vs. multi-product;
- the problem temporal resolution, i.e., single- vs. multi-period;
- the supply chain depth in terms of number of levels and stages, i.e., from the two level producer-customer direct distribution to long indirect delivery channels with one or two intermediates;
- the feasible shipping modes to distribute products.

The second pattern (*cost drivers*) explores the costs rising during production and distribution activities. Though large differences exist among applications and industrial sectors, the most common drivers are:

- the direct production cost for manufacturing, assembling, inspection, packing and inbound handling;
- the shipping & handling cost for outbound logistics and delivery of products along the supply chain;
- the inventory and storage cost for stocking products, even using 3PL service providers;
- the opportunity cost, i.e. missing revenues, due to lost orders and unmet demand;
- the setup cost to reconfigure the production plants before each new batch.

Finally, about the third pattern (*constraints*), elements considered by the literature and making the problem consistent and realistic deal with:

- the inclusion of plant production capacity, up- and down-times and resource capacity constraints;
- the inclusion of local storage capacity at each level of the network;
- the possibility of 3P storage to manage peaks and seasonality effects;
- the inclusion of the effects of stock-outs and unmet demand;
- the inclusion of minimum lot sizes and batch policy management.

		Entity Dimensions			Cost Drivers					Constraints				
		Products/Items	**SC levels**	**Shipping mode**	**Manufacturing**	**Shipping & handling**	**Inventory**	**Opportunity cost**	**Setup costs**	**Resource capacity**	**Internal storage capacity**	**3P storage capacity**	**Stock-out management**	**Minimum production batches**
2007	[5]	M	3	S	X	X	X			X	X			
2008	[6]	M	4	M	X	X	X			X	X	X		
2012	[7]	M	3	S	X	X			X					X
2012	[8]	M	4	M	X	X	X			X	X	X		
2016	[9]	S	2	M	X	X			X	X	X			X
2016	[10]	M	3	M		X								
2017	[11]	M	3	S	X	X	X			X	X		X	
2017	[12]	S	2	S		X				X				
2017	[13]	S	3	M	X	X	X		X	X	X	X		X
2017	[14]	M	2	S	X	X	X	X	X	X	X		X	
2018	[15]	M	4	M	X	X			X	X	X			
2018	[3]	M	4	M	X	X	X			X	X			
2018	[16]	M	4	M	X	X				X	X			
2018	[17]	S	3	S	X	X	X			X				
2018	[18]	S	4	S	X	X	X	X		X	X	X		
2018	[19]	S	4	S	X	X	X			X	X			X
2019	[20]	M	3	S	X	X				X				
2019	[21]	M	4	S	X	X		X		X				X
2019	[22]	M	2	S	X	X			X	X				X
2019	[23]	S	3	S	X	X	X	X		X	X			
2019	[24]	M	4	S	X	X	X							

Table 1: Literature and classification of relevant recent contributions about production and distribution planning problems.

Table 1 revises the literature and classifies a major set of recent contributions according to the introduced patterns and relevant elements. The first two columns of the table report the year and reference of each contribution, respectively. The remaining columns refer to the characteristic elements of each contribution, classified according to the three above mentioned patterns, while each row gives the specific characteristics of the problem addressed in each contribution. To focus the analysis and for the sake of conciseness, only multi-period problems are considered. The table shows that a range of SC levels between two and four to distribute one or multiple products with one or more shipping modes. Concerning costs, despite some drivers are not negligible, e.g. production and distribution costs, others are considered rarely, e.g. opportunity and setup costs. Finally, the effects of stock-outs and batch policies are in a minor set of contributions. Though all aspects that are present in our problem have been addressed in the literature, to the best of our knowledge, there is no contribution from the literature in which all elements of patterns (ii) and (iii) in a multi-product, multi-mode and multi-level environment have been addressed.

Concerning solution strategies and algorithmic contributions, we identified the *lot-sizing problem* as the core problem of our topic. Given the amount of an independent demand, the lot-sizing is the problem of deciding which equipment must be used and in which period the production must be performed in order to satisfy the demand, while minimizing the total required cost. It has been addressed for the first time in 1913 in a seminal paper by Harris [25], who introduced the definition of *Economic Order Quantity* (EOQ). The vastness of the possible applications of this problem results evident from its definition; consequently, the number of possible variants is also huge. For these reasons, understanding which is the specific version of the lot-sizing problem that one is dealing with is a crucial task that is, at the same time, not straightforward. The tertiary study proposed by Glock et al. [26] results very useful for a first orientation. According to the definitions of this tertiary study, our case can be defined as an extension of the multi-item Capacitated Lot Sizing Problem (CLSP). This problem considers a discretized planning horizon and aims to determine the amount and production level in each period. Typically, a finite capacity bounds the production level in each period, and the problem involves production costs, set up costs, and holding costs. Production costs and holding costs are linear functions of the production level and of the number of product that are stored, respectively; on the contrary, set up costs are fixed cost that occur only when a machine starts or switches the production in some period. In particular, the problem we consider corresponds to a dynamic (demand is not constant) and deterministic (demand is known) variant of the basic probelm. The work of Karimi et al. [27] is indicated as one of the main reviews about this topic. The single-item version of the problem has been shown to be NP-hard by Florian et al. [28], even for the special case in which the demand is constant in every period and there are no holding costs; indeed, in this case the problem can be reduced to a feasibility checking for a 0-1 equality constrained knapsack problem. Furthermore, the multi-items version of the problem is known to be *strongly* NP-hard, as shown by Chen and Thizy [29]; in this case, the problem is reduced in pseudo-polynomial time, to the well-known Three Partition Problem.

Due to the complexity of the problem, there are few attempts in the literature that try to make use of an exact algorithm: for example, Armentano et val. [30] used a branch-and-bound procedure to solve the multi-item, single-level CLSP with setup times reformulated as a minimum cost network flow problem. On the other hand, there are many contributions based on heuristic approaches. According to [27], these approaches can classified in two different categories: common-sense heuristics and mathematical programming-based heuristics. The optimization algorithm that will be introduced in the next sections belongs to the former class. Using the terminology of [27], it includes both a *period-by-period* constructive heuristic (see Section 4), that builds a feasible solution by considering one demand at a time starting from the first period, and an *improvement* heuristic (see Section 5), that improves a given feasible solution. As for the works that also adopt this kind of procedures, in addition to the papers cited in [27], we also indicate the more recent studies of Li et al. [31] and of Toscano et al. [32]. As to mathematical programming-based heuristics, we mention the algorithm introduced by Tempelmeier [33]; this approach is based on a heuristic column generation procedure combined with a heuristic procedure, and is used to solve a dynamic multi-item capacitated lot-sizing problem under uncertain demands. As already mentioned, our problem can be defined as an extension of the CLSP. Compared with the basic version of the problem, several additional aspects appear in our application, namely: distribution costs (both from the company plants to the customers and from the company plants to external warehouses), setup times, stockout costs, availability of different capacity configurations, and minimum lot size. Most of these aspects appear separately in many papers proposing solution approaches but, to the best of our knowledge, there are no previous studies on problems that simultaneously involve all these aspects. The distribution costs were considered by Chandra and Fisher [34], who showed that considering the production and the distribution problems in an integrated system leads to better results than solving the two problems separately. Haq et al. [35] presented a case study where they formulated an integrated production-inventory-distribution model; similar integrated models were also proposed by Bhutta et al. [36] and by Jolayemi and Olorunniwo [37]. In all these cases the authors only provide a mathematical model to describe the problem at hand, without introducing any solution approach. Rao [38] introduced the capacity constraint for a single-item environment and solved the resulting problem using a dynamic programming algorithm. Rajagopalan and Swaminathan [39] considered the capacity constraint in a multiple item setting; for this problem, a Lagrangian relaxation procedure and two heuristic algorithms were proposed. Setup times were addressed by Jans and Degraeve [40] who proposed a branch-and-price algorithm, and by Hindi et al. [41], who proposed a smoothing heuristic procedure based on the Lagrangian relaxation of the problem. We also mention the dynamic multi-level capacitated lot sizing problem addressed by Chen [42], where the setup state of a resource may be used in consecutive time periods (*setup carryover*). The problem is solved using a variant of the fix-and-optimize approach introduced by Helber and Sahling [43], dividing the problem in a number of smaller subproblems that are iteratively solved in a heuristic way. Aksen et al. [44] addressed the lot-sizing problem with stockouts but where a single product is present. The authors analyzed several structural properties of optimal solutions of the model and, on the basis of these properties, derived a forward recursive dynamic programming algorithm. Finally, different problems including the minimum lot size constraint were addressed using optimization algorithms. Constantino [45] studied the problem from the polyhedral viewpoint, while Mercé and Fontan [46] solved the problem using a MILP-based heuristic.

3. Problem Statement

We consider the problem of a company that faces a multiperiod demand for the assortment of different products. The demand of each period originates from a set of customers with a geographic distribution, and the products can be obtained from different plants, from which they are shipped to the customers. Plants are equipped with production lines, each one with the capability to produce, in each period, a subset of the products after a suited setup; for each product, there is a minimum production quantity that can be considered. Production consumes resources that can be available at company, plant or line level. Production lines work according to shift configurations, that define the number of production hours during a period and that can be changed only at the beginning of some periods. Each plant is associated with an internal warehouse, where the production of a period can be stored before it is shipped to a customer at a later period. Each plant can also rent additional space from external warehouses, if needed. The company wants to decide, for each period, the shift configuration of each line and the production level of each product, in order to minimize the total cost, determined by configuration and setup costs for the lines, production, storage and shipment costs for the demands that are satisfied, and penalty costs for the demands that are not satisfied. A formal definition of the problem and a mathematical model are given in what follows.

3.1 Sets and variables

In our formulation we consider the following sets

- $\mathcal{K} = \{1, \ldots, K\}$: set of products;
- $\mathcal{J} = \{1, \ldots, J\}$: set of customers;
- $\mathcal{I} = \{1, \ldots, I\}$: set of plant;
- $\mathcal{M} = \{1, \ldots, M\}$: set of lines. Each line m is associated with a specific plant $i(m) \in \mathcal{I}$, that is, for each plant $i \in \mathcal{I}$ we define the set of associated lines as $M(i)$. Compatibility between products and lines is expressed defining, for each product $k \in \mathcal{K}$, a set $C(k)$ of lines that can produce k;
- $H(m)$: set of configurations for each line $m \in \mathcal{M}$. Each shift configuration for a line has associated a cost and determines the capacity of the line. Without loss of generality, we assume that a larger cost corresponds to a larger capacity, and that configurations are sorted by increasing cost;
- $\mathcal{F} = \{1, \ldots, f\}$: set of product families; each product k belongs to one family $f_k \in \mathcal{F}$;
- $\mathcal{T} = \{1, \ldots, T\}$: set of time intervals. Set $\mathcal{T}$ is partitioned into π subsets $\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_\pi$, that are called *multiperiods*. Each multiperiod includes all the time intervals between two consecutive periods in which line configurations can be changed;
- $\mathcal{RP}_i = \{1, \ldots, RP_i\}$: set of resources for plant i;

and decision variables specified as follows

- x_{kmt} = number of products k obtained on line m during period t $(k \in \mathcal{K}, m \in C(k), t \in \mathcal{T})$. These integer variables encode the main decision for the company, that is defining about production over time periods and lines;

- z_{kjt} = unsatisfied demand for product k at period t for customer j $(k \in \mathcal{K}, j \in \mathcal{J}, t \in \mathcal{T})$. These integer variables define the amount of customers demand that the company does not satisfy;

- f_{kjit} = amount of product k shipped from plant i to customer j at period t $(k \in \mathcal{K}, j \in \mathcal{J}, i \in \mathcal{I}, t \in \mathcal{T})$. These integer variables define the way in which customers demand is satisfied by the company;

- $y_{mhv} = \begin{cases} 1 & \text{if line } m \text{ operates in configuration } h \text{ during multiperiod } v \\ 0 & \text{otherwise} \end{cases}$
  $(m \in \mathcal{M}, h \in H(m), v = 1, \ldots, \pi)$.
  These binary variables define the shift configuration (working hours) of a line during each multiperiod v.

- w_{kit} = amount of product k stored in the internal warehouse associated with plant i during period t $(k \in \mathcal{K}, i \in \mathcal{I}, t \in \mathcal{T})$;

- s_{kit} = amount of product k stored in the external warehouse associated with plant i during period t; $(k \in \mathcal{K}, i \in \mathcal{I}, t \in \mathcal{T})$;

- ϕ_{kit} = amount of product k shipped from plant i to the associated external warehouse during period t $(k \in \mathcal{K}, i \in \mathcal{I}, t \in \mathcal{T})$;

- $\alpha_{mft} = \begin{cases} 1 & \text{if line } m \text{ processes at least one product belonging to family } f \text{ during period } t \\ 0 & \text{otherwise} \end{cases}$
  $(m \in \mathcal{M}, f \in \mathcal{F}, t \in \mathcal{T})$.
  These binary variables are used to denote the product families that are processed on a line during a specific period, and to allocate the associated setup costs.

3.2 Cost function

The total cost to be minimized takes into account different components: the costs for operating the lines, the production and operational costs for the satisfied demands, and the opportunity costs (penalties) for the demands that are not satisfied. In particular, the overall cost is defined as the sum of the following terms

C^{PR} is the direct production cost, defined as $C^{PR} = \sum_{k \in \mathcal{K}} \sum_{m \in \mathcal{M}} \sum_{t \in \mathcal{T}} c^{PR}_{km} x_{kmt}$, where c^{PR}_{km} is the cost for producing a unit of product k on line m;

C^{NS} is the opportunity cost for the unsatisfied demand, defined as $C^{NS} = \sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}} \sum_{t \in \mathcal{T}} c^{NS}_{kt} z_{kjt}$, where c^{NS}_{kt} is the opportunity cost for the unsatisfied demand of a unit of product k at period t;

C^{TC} is the shipping cost to the customers, defined as $C^{TC} = \sum_{k\in\mathcal{K}} \sum_{j\in\mathcal{J}} \sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}} c^{TC}_{kji} f_{kjit}$ where c^{TC}_{kji} is the shipping cost for a unit of product k to customer j from plant i or from an internal/external warehouse associated with the plant;

C^{DC} is the configuration cost for the product lines, defined as $C^{DC} = \sum_{m\in\mathcal{M}} \sum_{h\in H(m)} \sum_{v=1}^{\pi} |\mathcal{T}_v| c^{DC}_{mh} y_{mhv}$, where c^{DC}_{mh} is the shift cost for line m in configuration h;

C^{SI} is the storage cost in the internal warehouses, defined as $C^{SI} = \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}} c^{SI}_{ki} w_{kit}$, where c^{SI}_{ki} is the storage cost for a unit of product k in the internal warehouses at plant i;

C^{SE} is the storage cost in the external warehouses, defined as $C^{SE} = \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}} c^{SE}_{ki} w_{kit}$, where c^{SE}_{ki} is the storage cost for a unit of product k in the external warehouses at plant i;

C^{SU} is the setup cost, defined as $C^{SU} = \sum_{m\in\mathcal{M}} \sum_{f\in\mathcal{F}} \sum_{t\in\mathcal{T}} c^{SU}_{m} \alpha_{mft}$, where c^{SU}_{m} is the cost for a setup on line m.

C^{TE} is the shipping cost to the external warehouses, defined as $C^{TE} = \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}} c^{TE}_{ki} \phi_{kit}$, where c^{TE}_{ki} is the shipping cost for a unit of product k to the external warehouses at plant i.

3.3 Constraints

The production schedule is subject to a number of different constraints, that we present grouped according to their meaning and structure.

Demand

$$\sum_{i\in\mathcal{I}} f_{kjit} + z_{kjt} = d_{kjt} \qquad k\in\mathcal{K}, j\in\mathcal{J}, t\in\mathcal{T} \tag{1}$$

These constraints impose that the demand is either satisfied by shipping the required amount of products from some plant, or it is canceled and computed as missed demand. The demand is expressed in terms of an order d_{kjt} for a product k to be delivered at customer j at time period t. Observe that we allow solutions where a customer is served by shipping products from different plants.

Lot-sizing balancing constraints

$$\sum_{m\in M(i)} x_{kmt} + s_{ki,t-1} + w_{ki,t-1} = \sum_{j\in\mathcal{J}} f_{kjit} + s_{kit} + w_{kit} \qquad k\in\mathcal{K}, i\in\mathcal{I}, t\in\mathcal{T} \tag{2}$$

$$\sum_{k\in\mathcal{K}} w_{kit} \le W_i \qquad i\in\mathcal{I}, t\in\mathcal{T} \tag{3}$$

These constraints generalize the classical lot-sizing formulation to the case of multiple products, plants and warehouses. Constraints (2) impose, for each product, plant, and period, a balance between the availability of the product, given by the production and the stocked quantities, and the stocked quantities at the end of the period plus the quantity that is shipped to customers. Constraints (3) define, for each plant and period, the available capacity W_i of the internal warehouse associated with the plant. Note that there is no similar constraint for external warehouses, as we assume they have unbounded capacities.

Resources

$$\sum_{k \in \mathcal{K}} \tau_{km}\, x_{kmt} + \sum_{f \in \mathcal{F}} t_m^{SU} \alpha_{mft} \leq \sum_{h \in H(m)} Q_{mh}\, y_{mhv} \qquad m \in \mathcal{M}, v = 1, \ldots, \pi, t \in \mathcal{T}_v \tag{4}$$

$$\sum_{h \in H(m)} y_{mhv} = 1 \qquad m \in \mathcal{M}, v = 1, \ldots, \pi \tag{5}$$

$$\sum_{k \in \mathcal{K}} \sum_{m \in M(i)} b_{kmr}^P\, x_{kmt} \leq B_{irt}^P \qquad i \in \mathcal{I}, r \in \mathcal{RP}_i, t \in \mathcal{T} \tag{6}$$

These constraints limit the use of available lines and resources according to their availabilities.

Constraints (4) impose an upper bound on the maximum working time for each line m in each period t. Indeed, the working time of the line is determined by the production times τ_{km} for each product k obtained on the line plus the total setup time, which is equal to t_m^{SU} for each different family that is produced. The right-hand-side gives the availability of the line, where each coefficients Q_{mh} represents the available working time for line m and shift configuration h. These constraints are paired with constraints (5) that impose to select exactly one shift configuration for each line and multiperiod. Finally, constraints (6) describe the consumption of resources available at plant level and impose, for each plant, resource and period, that the amount of resource used cannot exceed the availability. In these constraints, each b_{kmr}^P is the amount of resource r used by line m to produce one product of type k, while B_{irt}^P denotes the availability of resource r at plant i during time period t. In a similar way, it is possible to express the constraints (if any) that regulate the consumptions of resources that are available at line or company level.

Setup, minimum production

$$\sum_{k \in \mathcal{K}: f_k = f} x_{kmt} \leq BIGM\, \alpha_{mft} \qquad m \in \mathcal{M}, t \in \mathcal{T}, f \in \mathcal{F} \tag{7}$$

$$(x_{kmt} = 0) \quad \vee \quad (x_{kmt} \geq \rho_{kmt}) \qquad k \in \mathcal{K}, m \in \mathcal{M}, t \in \mathcal{T} \tag{8}$$

Constraints (7) model the setup that are incurred for each additional line, period and product family. In these constraints, $BIGM$ is a large enough constant that is used to model the logical implication forcing an α variable to take value 1 in case some products belonging to a certain family are produced. Observe that we consider a planning problem, thus we do not model the specific setup time and cost as a function of the production sequence, and adopt instead average values for setup times and costs. Logical constraints (8) impose that, for each product, line, and period, either there is at least a minimum production level ρ_{kmt} of the product, or there is no production at all for that product.

Shipment to external warehouses

$$\phi_{kit} \geq s_{kit} - s_{ki,t-1} \qquad k \in \mathcal{K}, i \in \mathcal{I}, t \in \mathcal{T} \tag{9}$$

Finally, constraints (9) model the flow of products to the external warehouses, for each product, plant and period. In particular, a positive value for a ϕ variable corresponds to some increase in the amount of products that stocked in an external warehouse.

Mathematical formulation Finally the overall Mixed Integer Formulation reads

$$\min \; C^{PR} + C^{NS} + C^{DC} + C^{SI} + C^{SE} + C^{TE} + C^{TC} + C^{SU} \tag{10}$$

$$(1) - (9)$$

$$f_{kjit} \geq 0 \qquad k \in \mathcal{K}, j \in \mathcal{J}, i \in \mathcal{I}, t \in \mathcal{T} \tag{11}$$

$$z_{kjt} \geq 0 \qquad k \in \mathcal{K}, j \in \mathcal{J}, t \in \mathcal{T} \tag{12}$$

$$x_{kmt} \geq 0 \qquad k \in \mathcal{K}, m \in C(k), t \in \mathcal{T} \tag{13}$$

$$w_{kit} \geq 0 \qquad k \in \mathcal{K}, i \in \mathcal{I}, t \in \mathcal{T} \tag{14}$$

$$s_{kit} \geq 0 \qquad k \in \mathcal{K}, i \in \mathcal{I}, t \in \mathcal{T} \tag{15}$$

$$y_{mhv} \in \{0,1\} \qquad m \in M, h \in H(i), v = 1, \ldots, \pi \tag{16}$$

$$\alpha_{mft} \in \{0,1\} \qquad m \in M, f \in \mathcal{F}, t \in \mathcal{T} \tag{17}$$

$$\phi_{kit} \geq 0 \qquad k \in \mathcal{K}, i \in \mathcal{I}, t \in \mathcal{T} \tag{18}$$

where the objective function (10) is the sum of the costs, all variables associated with physical quantities are defined as continuous nonnegative, and activation variables associated with line configurations and setups are binary.

The model has a polynomial number of variables and constraints. Note that constraints (8) are not linear, in that they model a disjunctive argument. However, it is well known that they can be rewritten in an equivalent form that is linear with respect to the decision variables. This requires the addition to the model of binary variables and $BIGM$ coefficients, possibly leading to formulations that have a weak linear programming relaxation.

4. A constructive algorithm

In this section we present an iterative constructive heuristic that is used to produce a feasible solution for the problem; possibly, this solution may be used as a starting point for the metaheuristic algorithm that we describe in the next section.

Our algorithm has a *primal* nature, in that it starts with a feasible solution and maintains feasibility at each iteration. Initially, no production takes place and each line operates in each multiperiod according with the configuration having minimum cost. Then, the algorithm considers one demand at a time, and determines the best policy for the current demand, according to a greedy strategy. For the current demand, the algorithm determines the "best" line and time period in which production must take place, the amount of products to be produced, and updates the production schedule accordingly. In case some change of configuration is required, the new configuration is maintained for all time periods of the multiperiod.

Remind that each demand has associated a product, a customer, and a time period. Demands are initially sorted by increasing time period, breaking ties according to the type of product and by the customer. Let k, j and t denote the product, the customer and the period, respectively, associated with the current demand. Then, we consider all lines $m \in C(k)$ that are compatible with products of type k and all time periods $\tau \leq t$. For each pair (m, τ) we determine the associated *production level*, i.e., the maximum amount $e(km\tau)$ of product k that can be produced for satisfying the current order. The total cost for pair (m, τ) is determined by

- the direct production cost;
- the shipping cost to the customer;
- possibly, the setup cost;
- possibly, the cost for changing configuration;
- possibly, the storage cost and the shipping cost to external warehouses.

The first two costs are related with production and shipping of the products, respectively. The setup cost is incurred only in case no product belonging to familty f_k is scheduled on line m in time period τ. As to the configuration cost, we take into account only the incremental cost that is incurred when the configuration for line m has to be increased to the next step thus allowing additional capacity. In the score computation, only a fraction of this additional cost is taken into account, the value of the fraction being a parameter that will be denoted by Λ and will be described later. Finally, storage costs are considered in case $\tau < t$, and may be associated with costs for shipping products to an external warehouse in case not all products can be stored in the internal warehouse in all the time interval $[\tau, t]$.

Due to the imposed constraints, e.g., capacity of the line or minimum amount of production on the line, the production level varies for each pair (m, τ). Thus, to have a meaningful comparison, we score each pair (m, τ) according to the cost per unit of product, i.e., we divide the total cost by the production level $e(km\tau)$, and select the pair $(\overline{m}, \overline{t})$ with minimum score. If no line can produce product k in any time period $\tau \leq t$, or the cost per unit of product is larger than the cost cost related to the out-of-stock of product k, the current demand is refused.

In case $e(k\overline{m}\overline{t}) < d_{kjt}$ we define a new (dummy) order with demand $d'_{kjt} = d_{kjt} - e(k\overline{m}\overline{t})$ and iterate the process. Note that the requirement on the minimum production (constraints (8)) may impose a production that is strictly larger than d_{kjt}. In this case, some units of product have to be stored in the warehouse, and used in some subsequent time period (the associated storage costs being not taken into account in this phase).

The pseudo-code of the algorithm is given below:

Algorithm Greedy:
sort the demands by nondecreasing time interval, breaking ties by product and by customer;
for each demand **do**
determine the "best" line $\overline{m}$, period $\bar{t}$ and production level $e(k\overline{m}\bar{t})$;
produce $e(k\overline{m}\bar{t})$ units of product k on line $\overline{m}$ in period $\bar{t}$;
possibly update the configuration of line $\overline{m}$ in the multiperiod containing $\bar{t}$;
if $e(k\overline{m}\bar{t}) < d_{kjt}$ **then** iterate with a new demand $d'_{kjt} = d_{kjt} - e(k\overline{m}\bar{t})$;
if $e(k\overline{m}\bar{t}) > d_{kjt}$ **then** store $e(k\overline{m}\bar{t}) - d_{kjt}$ units of product k in the warehouse associated with the plant $i(\overline{m})$;
end for

5. A metaheuristic approach

Preliminary computational experiments showed that the constructive heuristic of Section 4 is very fast but may produce solutions in which the use of some lines is not fully optimized. Thus, we developed a metaheuristic approach that is based on the ruin-and-recreate paradigm (see, Schrimpf et al. [47]). The idea of a ruin-and-recreate algorithm is to determine feasible solutions by (i) destroying a considerable part of a feasible solution (*ruin* phase), and (ii) applying a rebuilding procedure (*recreate* phase) to complete the solution.

In our algorithm, the initial solution is produced by the constructive heuristic, and three different procedures are used to destroy the solution:

1. remove all production associated with a given line in a given multiperiod;

2. remove all production associated with a given product in all time periods;

3. remove all production associated with a given line in all time periods.

The first ruin procedure is used to repair situations in which the configuration of the line is not optimized in the first periods of a multiperiod. Typically this happens when a change of configuration is incurred at some intermediate period of the multiperiod, preventing all previous time intervals to take advantage of the increased capacity of the line. The procedure considers one pair (line m, multiperiod v) at a time, removes all the production of line m in multiperiod v, and repairs the solution through multiple runs of the constructive procedure of Section 4. In particular, at each run we guess the configuration for the line m in multiperiod v, and try to reschedule all the demands that are unsatisfied (either unscheduled in the initial solution or removed by the ruin phase) without changing configuration of the lines. The best solution found is used to replace the incumbent, in case it produces some improvement.

The second ruin procedure considers one product k at a time, removes the production of product k from all lines and periods, and reschedules unsatisfied demands by applying again the constructive heuristic (without changing the configuration of the lines).

Similarly, the third ruin procedure considers each line m at a time, removes all the production on the current line, and reschedules unsatisfied demands using the constructive heuristic (without changing the configuration of the lines).

It should be noted that, in all cases, the repair phase is obtained running the constructive procedure of Section 4 to a restricted problem in which the configuration of all lines in all the multiperiods is fixed. For this reason, configuration costs are not taken into account in this phase.

The pseudo-code of the metaheuristic algorithm is as follows:

Algorithm Ruin-and-Recreate:
determine an initial feasible solution using Algorithm **Greedy**;
repeat
    for each line m and multiperiod v **do**
        remove all production on line m during multiperiod v;
        for each configuration $h \in H(m)$:
            set configuration h for line m in multiperiod v;
            schedule unsatisfied demands without changing the configuration of the lines;
            possibly update the incumbent;
        end for
    end for
    for each product k **do**
        remove all production of type k;
        schedule unsatisfied demands without changing the configuration of the lines;
        possibly update the store the incumbent;
    end for
    for each line m **do**
        remove all production on line m;
        schedule unsatisfied demands without changing the configuration of the lines;
        possibly update the store the incumbent;
    end for
until (stopping conditions)

In our implementation, the algorithm is halted after a maximum number *IterNum* of iterations or when an iteration with no improvement is encountered.

6. Computational experiments

In this section we evaluate the computational performance of our approach on a set of real and realistic instances derived from two industrial problems in the food and beverage industry. The objective of our experiments is threefold:

- first, we want to tune the algorithm parameters and assess the performance of the two steps of our approach, in terms of solution quality and computing time;

- second, we want to compare the overall performance of our approach with the direct solution of mathematical model (10)–(18) through a state-of-the-art MILP solver;
- and finally, we are interested in analyzing the structure of the obtained solutions for the industrial problems in both the original case and under some realistic business scenarios.

Real-world test-cases. We considered two real case studies, denoted as A and B in the following, that describe real-world situations arising from the food industry. In these instances the demand to be satisfied is specified in terms of *orders*, where an order is given by an a request for an amount d_{kjt} of product k by customer j at period t.

The first case study comes from a Food&Beverage company, operating mainly on the Italian market. Demand is highly seasonal and influenced by climatic conditions during hot summer seasons. The production process is based on high-productivity lines, possibly requiring production stability because product changes heavily influence efficiency and scraps (to bring the line to speed). In this application, shipping costs have a relevant economical impact, mainly on the "less profitable" products, that are manufactured in 6 plants, located in Italy and focused on the national market. Foreign markets are served only from the main plant, where also the most profitable and complex products are produced. Indeed, this plant has the largest production capacity, and is capable to manage the widest production mix. In this industry, supply chain planning is currently organized into 2 different levels of responsibility, namely corporate planning and plant planning, possibly yielding to some inefficiency; the proposed approach allows an integrated solution of these two phases, and may produce considerable savings.

Case B is a global food/bakery company. Demand is less seasonal than in the previous case and is mainly influenced by the trends in lifestyle. The production process starts with semi-finished items produced in batches, that are transformed and packaged in lines. In this case, changeovers and setup operations considerably impact on manufacturing, as specific cleaning procedures need to be carried out in the shop-floor to eliminate potential allergenic contaminations. Despite operations are distributed into 9 different plants, focused on specific finished products categories, some of the high-volume items can be manufactured in alternative plants, to serve market demand in a more efficient way. Batch manufacturing of the semi-finished items is centralized in a few facilities, and the products are then distributed to the other manufacturing plants. In both cases, the planning managers have also to take into account all the issues connected to product shelf-life.

Table 2 reports the main characteristics of the two test-cases, expressed in terms of number of products ($|\mathcal{K}|$), customers ($|\mathcal{J}|$), plants ($|\mathcal{I}|$), lines ($|\mathcal{M}|$), different shift configu-

	$\|\mathcal{K}\|$	$\|\mathcal{J}\|$	$\|\mathcal{I}\|$	$\|\mathcal{M}\|$	$\|\mathcal{H}\|$	$\|\mathcal{F}\|$	$\|\mathcal{T}\|$	$\|\pi\|$	$\|\mathcal{O}\|$
A	887	3	6	40	5	160	41	11	27,760
B	541	7	9	31	7	152	52	14	26,024

Table 2: Main characteristics of the real-world test-cases.

rations ($|\mathcal{H}|$), product families ($|\mathcal{F}|$), time periods ($|\mathcal{T}|$) and multiperiods ($|\pi|$). Finally, the last column ($|\mathcal{O}|$) gives the total number of orders that have to be satisfied.

Realistic instances. Starting from the two real-world test-cases, we produced an additional benchmark of 80 *realistic* instances. The realistic instances were obtained by a random perturbation of the original input data. In particular, for each real-world test-case, we generated:

- 10 *similar* instances in which the demand of each order d_{kjt} has been replaced by a demand in the range $[0.9\, d_{kjt}, 1.1\, d_{kjt}]$;
- 10 *varied* instances in which the demand of each order d_{kjt} has been replaced by a demand in the range $[0.7\, d_{kjt}, 1.3\, d_{kjt}]$.

All the remaining input parameters were left unchanged.

Experimental setting. All procedures were implemented in C++ and all experiments were performed on a computer equipped with a 3.20 GHz Intel Core I7 processor and 64 GB of RAM. Our algorithm is sequential in nature and cannot take advantage from the parallelism of our hardware. However, considering the orders according to different sortings produces different solutions. Thus, we implemented a parallel version of the algorithm in which 4 threads are used, each running the same algorithm on the same input, but with a different sequence for processing the orders. Then, the best of the four solutions is returned to the user. Preliminary experiments in which orders are considered according to (common-sense) deterministic rules produced solutions of the same quality as using random sortings.

6.1 Parameter Tuning

Our solution approach includes 2 main parameters:

- Parameter Λ, which defines the fraction of additional cost for the activation of a shift configuration that is allocated to the current order;
- Parameter *IterNum*, which determines the number of iterations of local search to be performed after the first phase of the algorithm.

We performed the tuning of our algorithm on the original instances using 7 different values for Λ, namely $0.00, 0.05, 0.10, 0.20, 0.30, 0.40$ and 0.50. Our first order of business was to determine the best values of parameter Λ for the two real-world test-cases. To this aim, we considered only the values of the solutions produced by the algorithm, without considering the computing time. Table 3 reports the total cost for different values of Λ and gives the best solution found by the 4-threads algorithm after the execution of the constructive algorithm and one iteration of Local Search. All entries in the table are given in M€, as the tactical nature of the problem, spanning over one year or alike, makes the solution cost very large. It is worth underlining that parameter Λ is used during the first phase of the algorithm only; thus, there would not be any additional information in considering more than one iteration of local search.

	0.00	0.05	0.10	0.20	0.30	0.40	0.50
A	119.15	118.43	118.17	118.29	119.57	120.09	122.62
B	263.44	252.66	248.13	255.64	265.97	280.48	288.84

Table 3: Effect of parameter Λ on real-world instances.

Results in Table 3 suggest that $\Lambda = 0.10$ is the best value for the two real-world test-cases. For the first instance, while the average solution value (for the considered values of the parameter) is equal to 119.47, using $\Lambda = 0.10$ produces a solution with value 118.17, with a saving equal to 1.09%. In the second case, the saving is more relevant and is equal to 6.37%.

As to the second parameter, we performed the following experiment. We considered instance A and run the algorithm for a (potentially very large) number of iterations, halting the procedure when no improvement is obtained after the last iteration of local search. Figure 1 reports the cost of the best solution found as a function of the computing time. The first point of the curve corresponds to the solution at the end of the first phase, while each subsequent point is the solution cost obtained after an additional iteration of the local search. As most of the computing time is spent in the local search phase, the computational effort grows almost linearly with the number of local search iterations. The picture shows that the first two iterations of the local search are very effective in reducing the solution cost, the associated cost reduction being around 9%. On the other hand, while subsequent iterations have a marginal effect. Thus, we set the second parameter *IterNum* to 2 in our

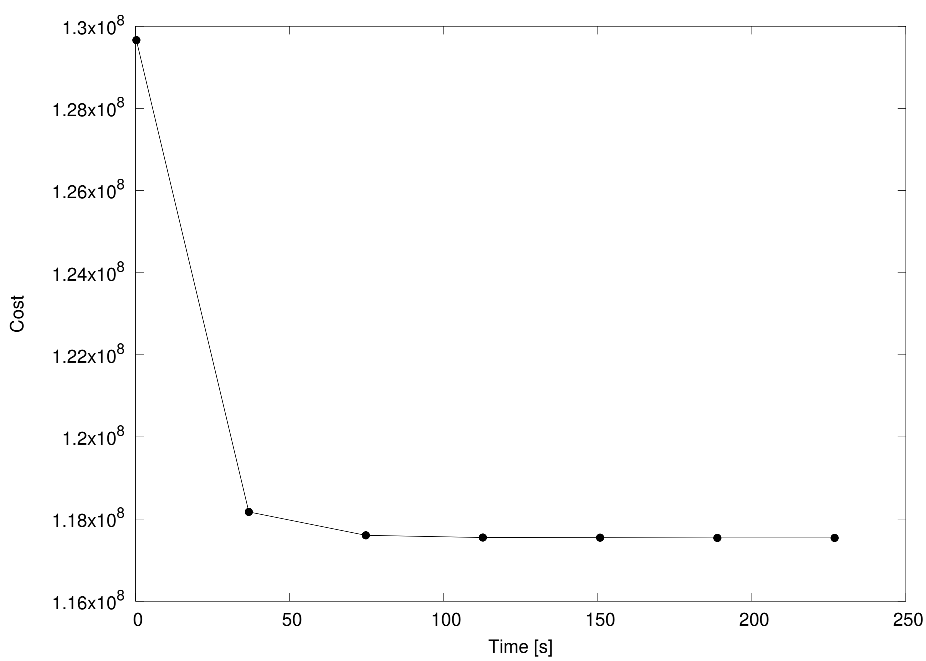

Figure 1: Cost of a solution as a function of the computing time for subsequent iterations of Local Search.

experiments.

For all the results reported in the remainder of the paper, the computational time with this configuration of the algorithm is approximately 90 seconds, for instance A, and 160 seconds, for instance B.

6.2 Comparison with MIP

In this section we compare the performances of our algorithm (in its best configuration, as described in the previous section) with those of a general-purpose commercial MILP solver. To this aim, we considered both the real-world and the realistic instances, and used IBM-ILOG Cplex 12.7.1 (CPLEX, in the following) to solve the mathematical formulation of Section 3.3. CPLEX was run on 4 threads with a limit of 10 hours of computing time; all the remaining parameters are set to their default values.

Table 4 compares the values of the gap obtained through our heuristic algorithm (HEUR, in the following) with those obtained by CPLEX. The first line of the table is associated with the real-world instances, while the remaining lines refer to the realistic instances. For each instance and algorithm we report the percentage *gap* of the best solution found, computed as $\%gap = 100\frac{U-L}{L}$, where U is the solution value and L denotes the best lower bound obtained by CPLEX at the time. In one instance, CPLEX could not terminate for memory reasons; in this case, marked with an asterisk (*), we report the gap reached until that moment. Due to the difficulty and size of the MILP formulation, we removed constraints (8) when solving the model with CPLEX. Thus, the results reported in columns CPLEX refer to solutions that are likely to be infeasible from this point of view, i.e., the comparison is by design in favour of CPLEX.

The results show that, on the real-world instances, our metaheuristic algorithm produces solutions whose gap is very close to that of the best solution found by CPLEX with a huge computational effort. Results are similar when realistic instances are considered, though CPLEX does not take the minimum production constraint into account.

6.3 Solution Structure and Scenario analysis

Finally, in this section we discuss the cost structure of the solutions computed by our method and we discuss how this structure changes under possible industrial scenarios. Figure 2 considers test-case A and compares the *as-is* solution (column 1) with the following *what-if* situations:

2. the use of external warehouses is forbidden;
3. the distribution costs are doubled;
4. the second most important plant is closes;
5. the scarce resource of the plant is reduced to the 70% of its value;
6. the scarce resource of the plant is available in an infinite quantity;
7. all the operation times of the lines are reduced to the 90% of their value.

Table 4: Comparison of the percentage optimality gap obtained by our algorithm (`HEUR`) and `CPLEX`

		A		B	
Type	Problem	`HEUR`	`CPLEX`	`HEUR`	`CPLEX`
-	-	12.92	9.11	15.98	16.16
Similar	1	13.08	10.45	16.65	16.18
	2	12.88	11.04	16.34	12.22
	3	12.79	10.20	16.78	15.07
	4	13.06	11.80	16.17	15.38
	5	12.82	9.35	16.33	17.85
	6	13.08	12.78	16.51	14.04
	7	12.95	10.66	16.68	16.47
	8	13.16	14.45	16.39	19.58
	9	13.19	10.12	16.38	13.87
	10	12.88	10.87	16.27	15.35
Varied	1	13.02	10.88	16.40	*19.93
	2	13.10	12.26	16.45	16.30
	3	13.59	11.18	16.86	15.75
	4	13.41	10.09	15.93	13.96
	5	13.26	10.28	16.24	15.74
	6	13.32	26.51	16.36	16.34
	7	13.40	12.40	17.04	14.36
	8	12.81	11.89	16.48	15.77
	9	13.40	9.41	16.94	13.96
	10	12.45	9.65	16.81	13.31

In scenario 2 the total cost is slightly larger than in the as-is solution; since the external warehouses are not available, the inventory costs decreases, but the amount of undelivered orders increases. In scenario 3, though distribution costs are doubled, the amount of orders that are delivered is almost the same as in the default setting; indeed, the Out of Stock costs do not change significantly with respect to the as-is situation, while the total distribution costs are actually doubled. Scenario 4 shows that closing an important plant has a relevant (negative) impact on the Out of Stock costs. Scenarios 5 and 6 show the importance of the plant resource availability: when it is decreased (scenario 5), then a larger part of the demand remains unsatisfied. If instead the resource is assumed to be unlimited (scenario 6), then the production is able to meet more than half of the demand that was unsatisfied in the as-is solution. Finally, scenario 7 shows that decreasing the operation times of the lines could allow to save a relevant fraction of the production and distribution costs. Observe that in all the scenarios, production costs are very small if compared with the other costs, confirming that is this beverage application shipping or configuration costs are the most relevant.

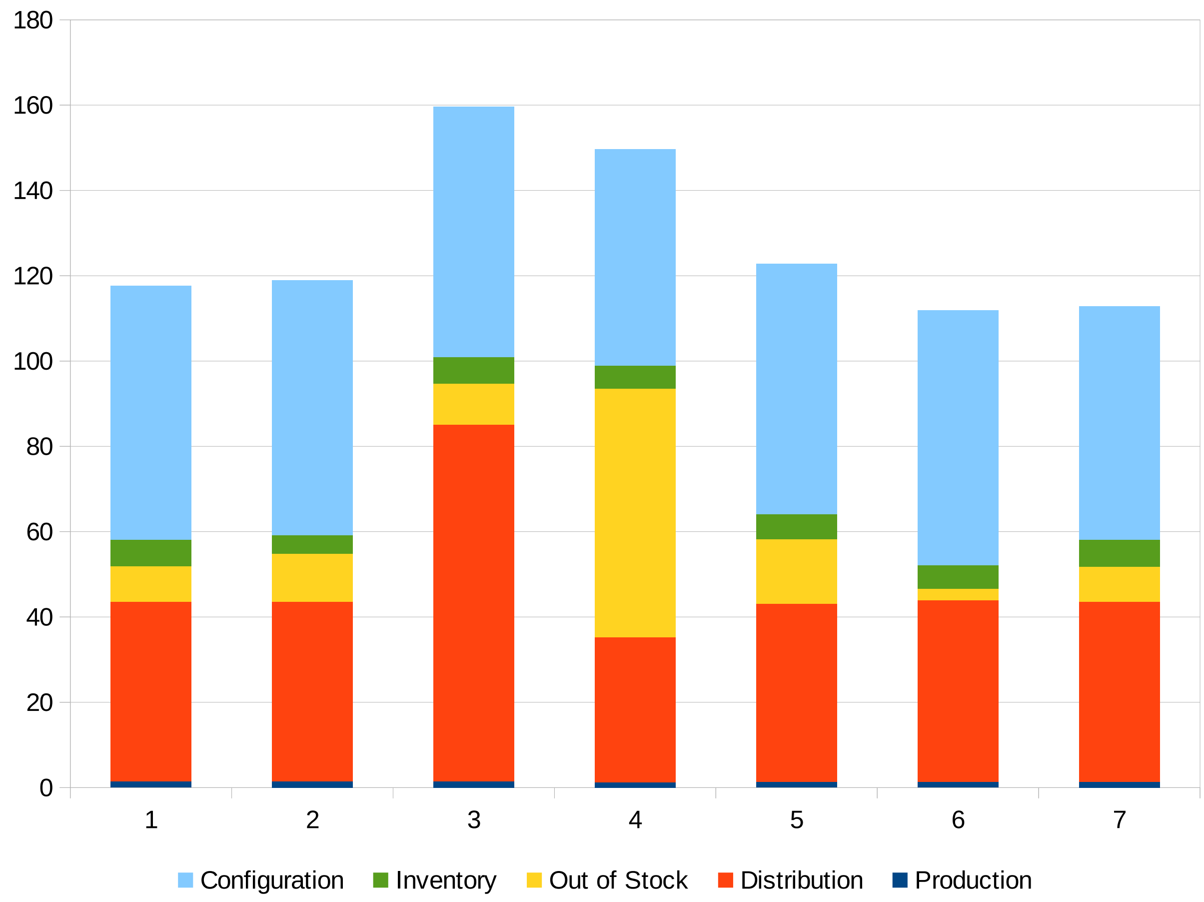

Figure 2: Cost distribution in different *what-if* scenarios.

7. Conclusions

In this paper we addressed an integrated production and distribution problem that arose from a real-world application. The problem includes several products, whose production must be scheduled on lines belonging to different plants, and several customers, whose demand must be satisfied taking into account the transportation costs from the plants to the customer locations.

After providing the formal definition of the problem based on a MILP formulation, we introduced both an iterative constructive heuristic algorithm and a metaheuristic approach based on the ruin-and-recreate paradigm.

An extensive analysis of the results of computational experiments on two real-world test-cases and on a set of realistic instances proved that the approximations obtained in short computing time by our metaheuristic approach are very close to those that can be achieved by running a state-of-the-art commercial solver on the mathematical model for a very long computing time.

Finally, our metaheuristic algorithm is used to perform a what-if analysis, aimed at determining the impact of different industrial scenarios on the solution cost.

Acknowledgments

The research has been partially supported by Regione Emilia-Romagna, under the POR-FESR 2014-20 project “PreLeveling & Plan Cost Optimization (PCO)”.

References

[1] B. Fahimnia, R. Z. Farahani, R. Marian, and L. Luong, “A review and critique on integrated production–distribution planning models and techniques,” *Journal of Manufacturing Systems*, vol. 32, pp. 1–19, 2013.

[2] M. Melo, S. Nickel, and F. Saldanha-da Gama, “Facility location and supply chain management – a review,” *European Journal of Operational Research*, vol. 196, pp. 401–412, 2009.

[3] M. Bortolini, F. G. Galizia, C. Mora, L. Botti, and M. Rosano, “Bi-objective design of fresh food supply chain networks with reusable and disposable packaging containers,” *Journal of cleaner production*, vol. 184, pp. 375–388, 2018.

[4] C. T. Maravelias and C. Sung, “Integration of production planning and scheduling: Overview, challenges and opportunities,” *Computers & Chemical Engineering*, vol. 33, pp. 1919–1930, 2009.

[5] R. A. Aliev, B. Fazlollahi, B. Guirimov, and R. R. Aliev, “Fuzzy-genetic approach to aggregate production–distribution planning in supply chain management,” *Information Sciences*, vol. 177, no. 20, pp. 4241–4255, 2007.

[6] H. Selim, C. Araz, and I. Ozkarahan, "Collaborative production–distribution planning in supply chain: A fuzzy goal programming approach," *Transportation Research Part E: Logistics and Transportation Review*, vol. 44, no. 3, pp. 396–419, 2008.

[7] P. Amorim, H.-O. Günther, and B. Almada-Lobo, "Multi-objective integrated production and distribution planning of perishable products," *International Journal of Production Economics*, vol. 138, no. 1, pp. 89–101, 2012.

[8] R. Manzini, "A top-down approach and a decision support system for the design and management of logistic networks," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 6, pp. 1185–1204, 2012.

[9] Y Boutarfa, A Senoussi, N. Mouss, and N Brahimi, "A tabu search heuristic for an integrated production-distribution problem with clustered retailers," *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 1514–1519, 2016.

[10] M. Hajghasem and A. S. Abbas, "Optimal routing in supply chain aimed at minimizing vehicle cost and supply," *Procedia Economics and Finance*, vol. 36, pp. 353–362, 2016.

[11] A. Chatzikontidou, P. Longinidis, P. Tsiakis, and M. C. Georgiadis, "Flexible supply chain network design under uncertainty," *Chemical Engineering Research and Design*, vol. 128, pp. 290–305, 2017.

[12] P. Devapriya, W. Ferrell, and N. Geismar, "Integrated production and distribution scheduling with a perishable product," *European Journal of Operational Research*, vol. 259, no. 3, pp. 906–916, 2017.

[13] M. Lai, X. Cai, and X. Li, "Mechanism design for collaborative production-distribution planning with shipment consolidation," *Transportation Research Part E: Logistics and Transportation Review*, vol. 106, pp. 137–159, 2017.

[14] W. Wei, L. Guimarães, P. Amorim, and B. Almada-Lobo, "Tactical production and distribution planning with dependency issues on the production process," *Omega*, vol. 67, pp. 99–114, 2017.

[15] H. Allaoui, Y. Guo, A. Choudhary, and J. Bloemhof, "Sustainable agro-food supply chain design using two-stage hybrid multi-objective decision-making approach," *Computers & Operations Research*, vol. 89, pp. 369–384, 2018.

[16] E. Dehghan, M. S. Nikabadi, M. Amiri, and A. Jabbarzadeh, "Hybrid robust, stochastic and possibilistic programming for closed-loop supply chain network design," *Computers & Industrial Engineering*, vol. 123, pp. 220–231, 2018.

[17] M. Du, N. Kong, X. Hu, and L. Zhao, "Tactical production and distribution planning in urban logistics under vehicle operational restrictions," *Procedia Computer Science*, vol. 126, pp. 1720–1729, 2018.

[18] A. Polo, N. Peña, D. Muñoz, A. Cañón, and J. W. Escobar, "Robust design of a closed-loop supply chain under uncertainty conditions integrating financial criteria," *Omega*, vol. 88, pp. 110–132, 2019.

[19] H. Rafiei, F. Safaei, and M. Rabbani, "Integrated production-distribution planning problem in a competition-based four-echelon supply chain," *Computers & Industrial Engineering*, vol. 119, pp. 85–99, 2018.

[20] E. Bottani, T. Murino, M. Schiavo, and R. Akkerman, "Resilient food supply chain design: Modelling framework and metaheuristic solution approach," *Computers & Industrial Engineering*, 2019.

[21] C. Gambella, F. Maggioni, and D. Vigo, "A stochastic programming model for a tactical solid waste management problem," *European Journal of Operational Research*, vol. 273, no. 2, pp. 684–694, 2019.

[22] J.-H. Han, J.-Y. Lee, and Y.-D. Kim, "Production planning in a two-level supply chain for production-time-dependent products with dynamic demands," *Computers & Industrial Engineering*, 2019.

[23] D. Han, Y. Yang, D. Wang, T. Cheng, and Y. Yin, "Integrated production, inventory, and outbound distribution operations with fixed departure times in a three-stage supply chain," *Transportation Research Part E: Logistics and Transportation Review*, vol. 125, pp. 334–347, 2019.

[24] M. Rahimi, V. Ghezavati, and F. Asadi, "A stochastic risk-averse sustainable supply chain network design problem with quantity discount considering multiple sources of uncertainty," *Computers & Industrial Engineering*, vol. 130, pp. 430–449, 2019.

[25] F. Harris, "How many parts to make at once," *Factory, the Magazine of Management*, vol. 10, pp. 135–136, 1913.

[26] C. H. Glock, E. H. Grosse, and J. M. Ries, "The lot sizing problem: A tertiary study," *International Journal of Production Economics*, vol. 155, pp. 39–51, 2014.

[27] B. Karimi, S. F. Ghomi, and J. Wilson, "The capacitated lot sizing problem: A review of models and algorithms," *Omega*, vol. 31, no. 5, pp. 365–378, 2003.

[28] M. Florian, J. K. Lenstra, and A. Rinnooy Kan, "Deterministic production planning: Algorithms and complexity," *Management science*, vol. 26, no. 7, pp. 669–679, 1980.

[29] W.-H. Chen and J.-M. Thizy, "Analysis of relaxations for the multi-item capacitated lot-sizing problem," *Annals of operations Research*, vol. 26, no. 1, pp. 29–72, 1990.

[30] V. A. Armentano, P. M. França, and F. M. de Toledo, "A network flow model for the capacitated lot-sizing problem," *Omega*, vol. 27, no. 2, pp. 275–284, 1999.

[31] Y. Li, Y. Tao, and F. Wang, "An effective approach to multi-item capacitated dynamic lot-sizing problems," *International Journal of Production Research*, vol. 50, no. 19, pp. 5348–5362, 2012.

[32] A. Toscano, D. Ferreira, and R. Morabito, "A decomposition heuristic to solve the two-stage lot sizing and scheduling problem with temporal cleaning," *Flexible Services and Manufacturing Journal*, vol. 31, no. 1, pp. 142–173, 2019.

[33] H. Tempelmeier, "A column generation heuristic for dynamic capacitated lot sizing with random demand under a fill rate constraint," *Omega*, vol. 39, no. 6, pp. 627–633, 2011.

[34] P. Chandra and M. L. Fisher, "Coordination of production and distribution planning," *European Journal of Operational Research*, vol. 72, no. 3, pp. 503–517, 1994.

[35] A. N. Haq, P. Vrat, and A. Kanda, "An integrated production-inventory-distribution model for manufacture of urea: A case," *International Journal of production economics*, vol. 25, no. 1-3, pp. 39–49, 1991.

[36] K. S. Bhutta, F. Huq, G. Frazier, and Z. Mohamed, "An integrated location, production, distribution and investment model for a multinational corporation," *International Journal of Production Economics*, vol. 86, no. 3, pp. 201–216, 2003.

[37] J. K. Jolayemi and F. O. Olorunniwo, "A deterministic model for planning production quantities in a multi-plant, multi-warehouse environment with extensible capacities," *International Journal of Production Economics*, vol. 87, no. 2, pp. 99–113, 2004.

[38] M. Rao, "Optimal capacity expansion with inventory," *Operations Research*, vol. 24, no. 2, pp. 291–300, 1976.

[39] S. Rajagopalan and J. M. Swaminathan, "A coordinated production planning model with capacity expansion and inventory management," *Management Science*, vol. 47, no. 11, pp. 1562–1580, 2001.

[40] Z. Degraeve and R. Jans, "A new dantzig-wolfe reformulation and branch-and-price algorithm for the capacitated lot-sizing problem with setup times," *Operations research*, vol. 55, no. 5, pp. 909–920, 2007.

[41] K. S. Hindi, K. Fleszar, and C. Charalambous, "An effective heuristic for the clsp with set-up times," *Journal of the Operational Research Society*, vol. 54, no. 5, pp. 490–498, 2003.

[42] H. Chen, "Fix-and-optimize and variable neighborhood search approaches for multi-level capacitated lot sizing problems," *Omega*, vol. 56, pp. 25–36, 2015.

[43] S. Helber and F. Sahling, "A fix-and-optimize approach for the multi-level capacitated lot sizing problem," *International Journal of Production Economics*, vol. 123, no. 2, pp. 247–256, 2010.

[44] D. Aksen, K. Altınkemer, and S. Chand, "The single-item lot-sizing problem with immediate lost sales," *European Journal of Operational Research*, vol. 147, no. 3, pp. 558–566, 2003.

[45] M. Constantino, "Lower bounds in lot-sizing models: A polyhedral study," *Mathematics of Operations Research*, vol. 23, no. 1, pp. 101–118, 1998.

[46] C. Mercé and G. Fontan, "Mip-based heuristics for capacitated lotsizing problems," *International Journal of Production Economics*, vol. 85, no. 1, pp. 97–111, 2003.

[47] G. Schrimpf, J. Schneider, H. Stamm-Wilmbrandt, and G. Dueck, "Record breaking optimization results using the ruin and recreate principle," *Journal of Computational Physics*, vol. 159, pp. 545–566, 2 2016.